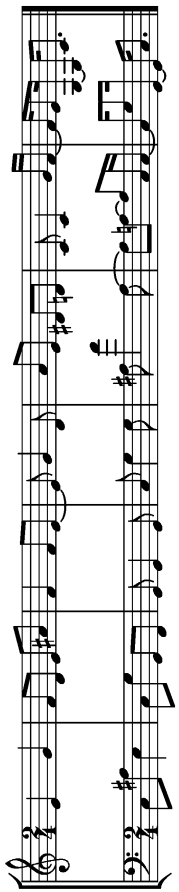
Probabilistic-logical Modeling of Music

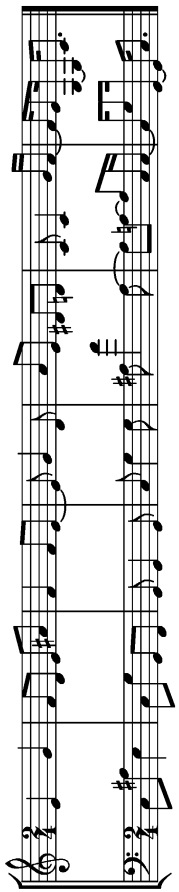
Jon Sneyers, Joost Vennekens, Danny De Schreye

PADL, January 9, 2006

Overview



Overview



Music genres

- Music composers restrict themselves to a genre
- Music genre is based on certain (unwritten) rules
- Many formalisms exist for strict rules (e.g. FOL)
- However: usually genre rules are not strict
- Solution: a probabilistic approach (using PRISM)

Music models

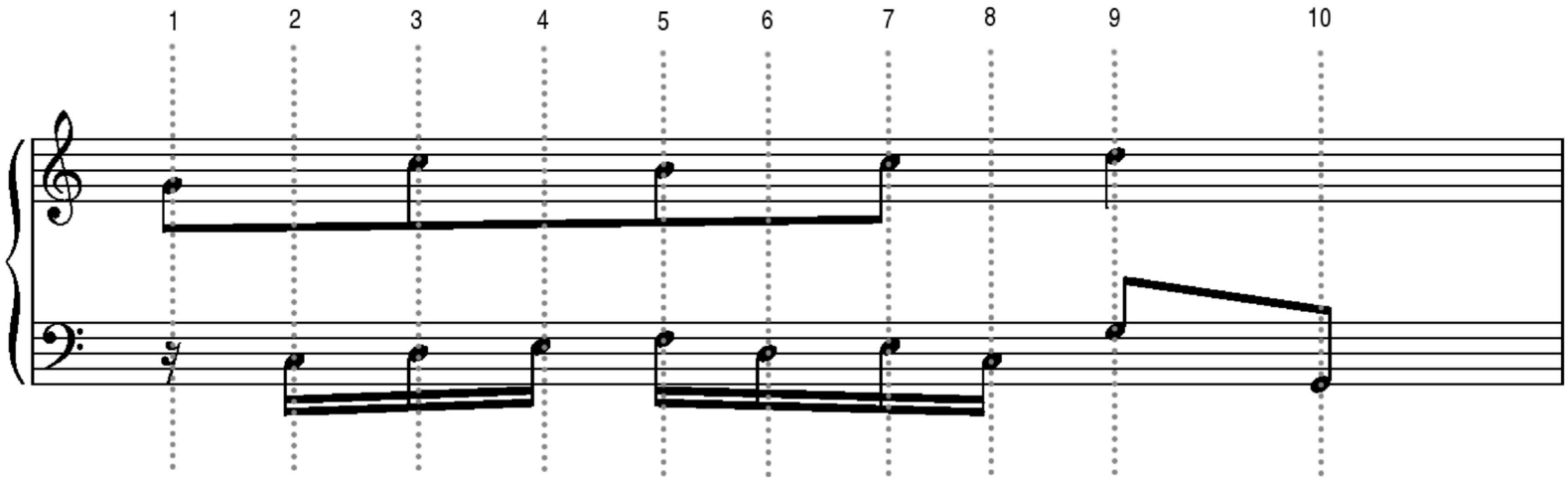
- Traditional distinction:
  - Synthetic models: automatic composition
  - Analytic models: classification
- General model for both tasks:
  statistical analytic model can be sampled
- Our goal was to design a model...
  - that can handle polyphonic music (many voices)
  - that automatically learns probability parameters
  - that is general (analytic and synthetic)

What to represent?

- "Essential" aspects :
  - Melody: note pitch/octave
  - Rhythm: note duration
  - Polyphony: voices sounding together
- Ignored aspects:
  - Dynamics: note volume
  - Instrumentation
  - Articulations (accents, staccato vs. legato, ...)

How to represent?

Interwoven Voices List



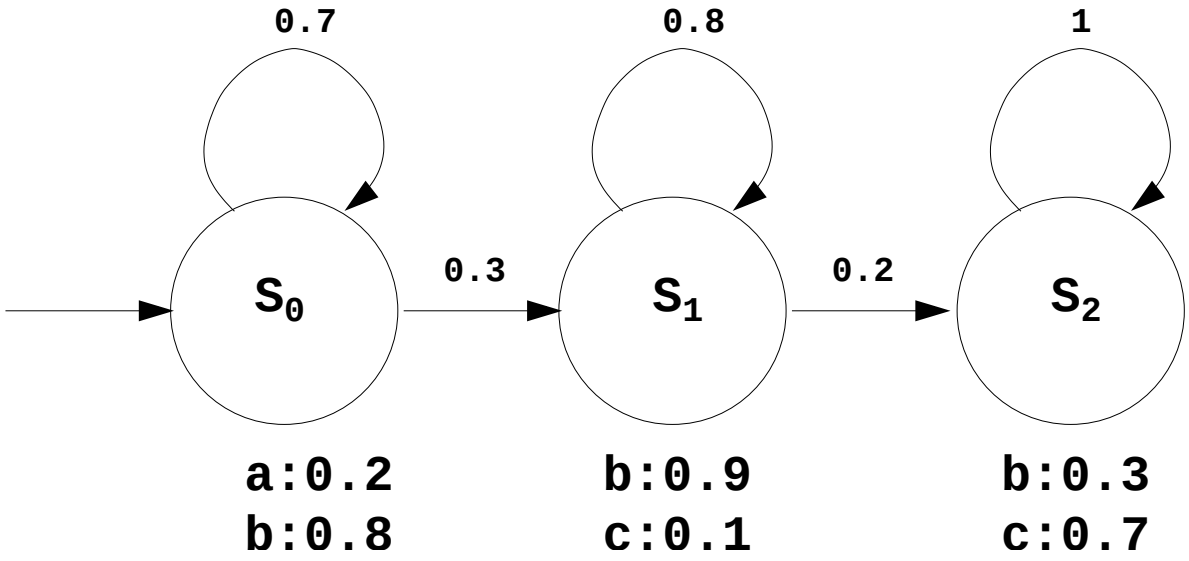

```
[ (16,[new,new],[55,r]), (16,[old,new],[55,36]),
  (16,[new,new],[60,38]), (16,[old,new],[60,40]),
  (16,[new,new],[59,41]), (16,[old,new],[59,38]),
  (16,[new,new],[60,40]), (16,[old,new],[60,36]),
  (32,[new,new],[62,43]), (32,[old,new],[62,31]) ]
```

PRISM

- PRISM: Programming In Statistical Modeling (T. Sato)
- PRISM = Prolog + probabilistic built-ins
- Use declarative LP to build complex models from basic probabilistic experiments
- Includes an efficient built-in EM learning algorithm
- PRISM program can be used for
  - Analysis: `prob/2` computes probability of some observation
  - Synthesis: `sample/1` generates an observation according to the probability distributions

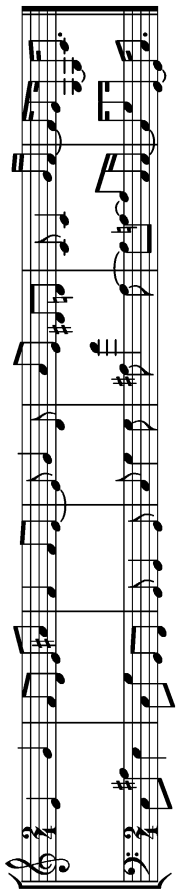
Hidden Markov Models

- a simple HMM:



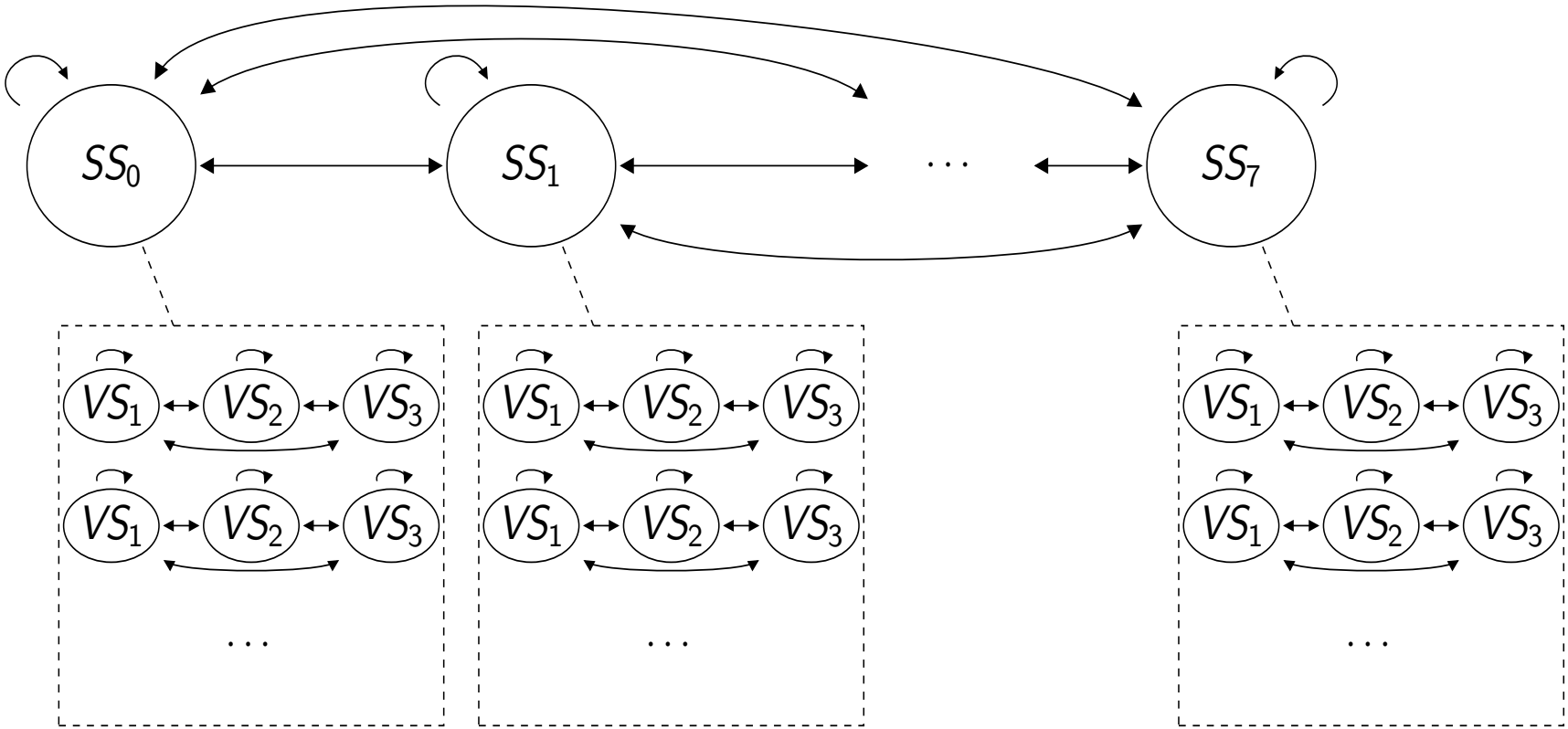
- HMM's can be modeled using PRISM
- PRISM's built-in learning algorithm is as efficient as special-purpose learning algorithms for HMM's

Overview

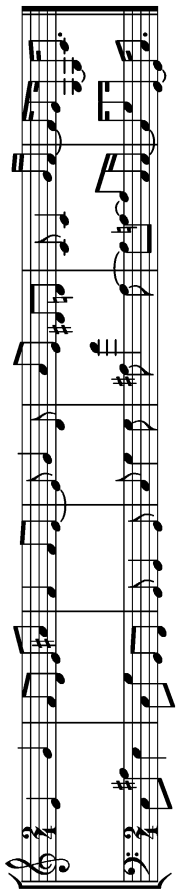


Music Model

- Global and local structure: *nested HMM*
- Song state / Voice states



Overview



Classification method

- Train separate instances of the model for every class (composer)
- How to classify an unknown piece X?
  - For every model, compute probability of X
  - Pick class corresponding to model with highest probability

Classification results

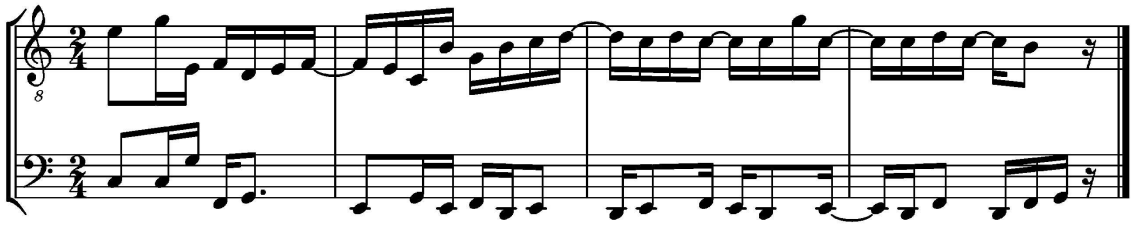
- We used 72 training fragments for Bach and 50 for Mozart
- All 30 unknown fragments are classified correctly
- . . . of which 20 have 3 voices while the training fragments all have 2
- When ignoring rhythm (durations), 26/30 are classified correctly

Generation method

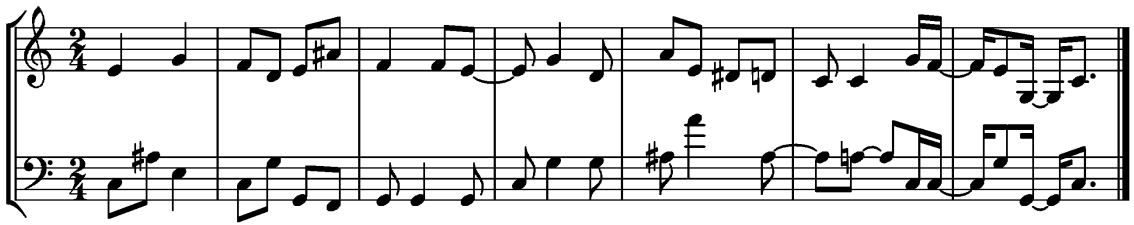
- Train the model with some training set
- Sample the model to get "typical" output
- We can choose the number of voices, song length and first notes

Generation results

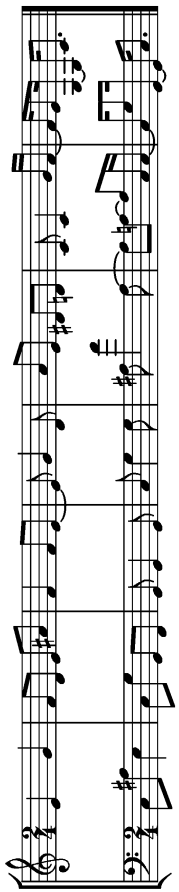
- Trained with Bach, two voices, 30 IVL phases:



- Trained with Mozart, two voices, 30 IVL phases:



Overview



Conclusion

- We have constructed a music model in PRISM
- it can handle polyphonic music
- it automatically learns probability parameters
- it can be used for synthesis and analysis
- Only took about 150 lines of PRISM code + some conversion scripts
- Nice application of probabilistic-logical programming

Future work

- Refinine the model to include more musical concepts (e.g. meter)
- Try more structured music representations
- More thorough experimental evaluation
- Compare with C(L)P-based techniques
- Maybe combine both approaches? (e.g. using probabilistic CHR)